# 8088
# 8-BIT HMOS MICROPROCESSOR
# 8088/8088-2

- 8-Bit Data Bus Interface
- 16-Bit Internal Architecture
- Direct Addressing Capability to 1 Mbyte of Memory
- Direct Software Compatibility with 8086 CPU
- 14-Word by 16-Bit Register Set with Symmetrical Operations
- 24 Operand Addressing Modes
- Byte, Word, and Block Operations
- 8-Bit and 16-Bit Signed and Unsigned Arithmetic in Binary or Decimal, Including Multiply and Divide
- Two Clock Rates:
  — 5 MHz for 8088
  — 8 MHz for 8088-2
- Available in EXPRESS
  — Standard Temperature Range
  — Extended Temperature Range

The Intel 8088 is a high performance microprocessor implemented in N-channel, depletion load, silicon gate technology (HMOS-II), and packaged in a 40-pin CERDIP package. The processor has attributes of both 8- and 16-bit microprocessors. It is directly compatible with 8086 software and 8080/8085 hardware and peripherals.

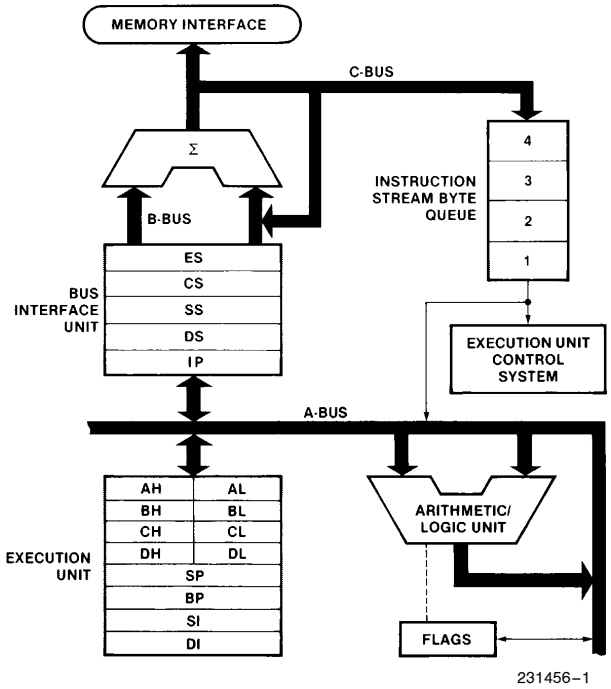



Figure 1. 8088 CPU Functional Block Diagram

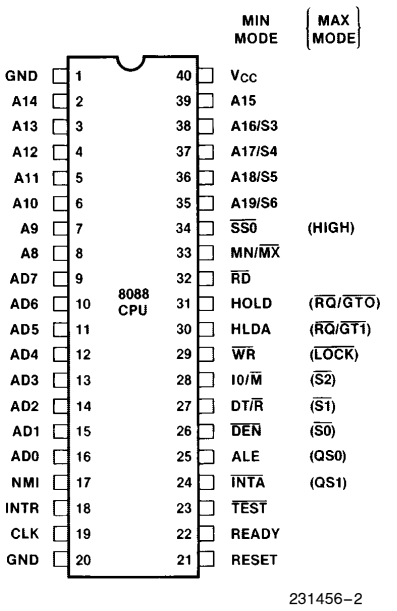



Figure 2. 8088 Pin Configuration

**Table 1. Pin Description**

*The following pin function descriptions are for 8088 systems in either minimum or maximum mode. The ''local bus'' in these descriptions is the direct multiplexed bus interface connection to the 8088 (without regard to additional bus buffers).*

| Symbol | Pin No. | Type | Name and Function |
|---|---|---|---|
| AD7–AD0 | 9–16 | I/O | **ADDRESS DATA BUS:** These lines constitute the time multiplexed memory/IO address (T1) and data (T2, T3, Tw, T4) bus. These lines are active HIGH and float to 3-state OFF during interrupt acknowledge and local bus ''hold acknowledge''. |
| A15–A8 | 2–8, 39 | O | **ADDRESS BUS:** These lines provide address bits 8 through 15 for the entire bus cycle (T1–T4). These lines do not have to be latched by ALE to remain valid. A15–A8 are active HIGH and float to 3-state OFF during interrupt acknowledge and local bus ''hold acknowledge''. |
| A19/S6, A18/S5, A17/S4, A16/S3 | 35–38 | O | **ADDRESS/STATUS:** During T1, these are the four most significant address lines for memory operations. During I/O operations, these lines are LOW. During memory and I/O operations, status information is available on these lines during T2, T3, Tw, and T4. S6 is always low. The status of the interrupt enable flag bit (S5) is updated at the beginning of each clock cycle. S4 and S3 are encoded as shown.<br>This information indicates which segment register is presently being used for data accessing.<br>These lines float to 3-state OFF during local bus ''hold acknowledge''. |

| S4 | S3 | Characteristics |
|---|---|---|
| 0 (LOW) | 0 | Alternate Data |
| 0 | 1 | Stack |
| 1 (HIGH) | 0 | Code or None |
| 1 | 1 | Data |
| S6 is 0 (LOW) | | |

| | | | |
|---|---|---|---|
| $\overline{RD}$ | 32 | O | **READ:** Read strobe indicates that the processor is performing a memory or I/O read cycle, depending on the state of the IO/$\overline{M}$ pin or S2. This signal is used to read devices which reside on the 8088 local bus. $\overline{RD}$ is active LOW during T2, T3 and Tw of any read cycle, and is guaranteed to remain HIGH in T2 until the 8088 local bus has floated.<br>This signal floats to 3-state OFF in ''hold acknowledge''. |
| READY | 22 | I | **READY:** is the acknowledgement from the addressed memory or I/O device that it will complete the data transfer. The RDY signal from memory or I/O is synchronized by the 8284 clock generator to form READY. This signal is active HIGH. The 8088 READY input is not synchronized. Correct operation is not guaranteed if the set up and hold times are not met. |
| INTR | 18 | I | **INTERRUPT REQUEST:** is a level triggered input which is sampled during the last clock cycle of each instruction to determine if the processor should enter into an interrupt acknowledge operation. A subroutine is vectored to via an interrupt vector lookup table located in system memory. It can be internally masked by software resetting the interrupt enable bit. INTR is internally synchronized. This signal is active HIGH. |
| $\overline{TEST}$ | 23 | I | **$\overline{TEST}$:** input is examined by the ''wait for test'' instruction. If the $\overline{TEST}$ input is LOW, execution continues, otherwise the processor waits in an ''idle'' state. This input is synchronized internally during each clock cycle on the leading edge of CLK. |

**Table 1. Pin Description** (Continued)

| Symbol | Pin No. | Type | Name and Function |
|---|---|---|---|
| NMI | 17 | I | **NON-MASKABLE INTERRUPT:** is an edge triggered input which causes a type 2 interrupt. A subroutine is vectored to via an interrupt vector lookup table located in system memory. NMI is not maskable internally by software. A transition from a LOW to HIGH initiates the interrupt at the end of the current instruction. This input is internally synchronized. |
| RESET | 21 | I | **RESET:** causes the processor to immediately terminate its present activity. The signal must be active HIGH for at least four clock cycles. It restarts execution, as described in the instruction set description, when RESET returns LOW. RESET is internally synchronized. |
| CLK | 19 | I | **CLOCK:** provides the basic timing for the processor and bus controller. It is asymmetric with a 33% duty cycle to provide optimized internal timing. |
| $V_{CC}$ | 40 | | **$V_{CC}$:** is the +5V ±10% power supply pin. |
| GND | 1, 20 | | **GND:** are the ground pins. |
| MN/$\overline{MX}$ | 33 | I | **MINIMUM/MAXIMUM:** indicates what mode the processor is to operate in. The two modes are discussed in the following sections. |

*The following pin function descriptions are for the 8088 minimum mode (i.e., MN/$\overline{MX}$ = $V_{CC}$). Only the pin functions which are unique to minimum mode are described; all other pin functions are as described above.*

| Symbol | Pin No. | Type | Name and Function |
|---|---|---|---|
| IO/$\overline{M}$ | 28 | O | **STATUS LINE:** is an inverted maximum mode $\overline{S2}$. It is used to distinguish a memory access from an I/O access. IO/$\overline{M}$ becomes valid in the T4 preceding a bus cycle and remains valid until the final T4 of the cycle (I/O = HIGH, M = LOW). IO/$\overline{M}$ floats to 3-state OFF in local bus "hold acknowledge". |
| $\overline{WR}$ | 29 | O | **WRITE:** strobe indicates that the processor is performing a write memory or write I/O cycle, depending on the state of the IO/$\overline{M}$ signal. WR is active for T2, T3, and Tw of any write cycle. It is active LOW, and floats to 3-state OFF in local bus "hold acknowledge". |
| $\overline{INTA}$ | 24 | O | **INTA:** is used as a read strobe for interrupt acknowledge cycles. It is active LOW during T2, T3, and Tw of each interrupt acknowledge cycle. |
| ALE | 25 | O | **ADDRESS LATCH ENABLE:** is provided by the processor to latch the address into an address latch. It is a HIGH pulse active during clock low of T1 of any bus cycle. Note that ALE is never floated. |
| DT/$\overline{R}$ | 27 | O | **DATA TRANSMIT/RECEIVE:** is needed in a minimum system that desires to use a data bus transceiver. It is used to control the direction of data flow through the transceiver. Logically, DT/$\overline{R}$ is equivalent to $\overline{S1}$ in the maximum mode, and its timing is the same as for IO/$\overline{M}$ (T = HIGH, R = LOW). This signal floats to 3-state OFF in local "hold acknowledge". |
| $\overline{DEN}$ | 26 | O | **DATA ENABLE:** is provided as an output enable for the data bus transceiver in a minimum system which uses the transceiver. $\overline{DEN}$ is active LOW during each memory and I/O access, and for $\overline{INTA}$ cycles. For a read or $\overline{INTA}$ cycle, it is active from the middle of T2 until the middle of T4, while for a write cycle, it is active from the beginning of T2 until the middle of T4. $\overline{DEN}$ floats to 3-state OFF during local bus "hold acknowledge". |

**Table 1. Pin Description** (Continued)

| Symbol | Pin No. | Type | Name and Function |
|---|---|---|---|
| HOLD, HLDA | 31, 30 | I, O | **HOLD:** indicates that another master is requesting a local bus ''hold''. To be acknowledged, HOLD must be active HIGH. The processor receiving the ''hold'' request will issue HLDA (HIGH) as an acknowledgement, in the middle of a T4 or Ti clock cycle. Simultaneous with the issuance of HLDA the processor will float the local bus and control lines. After HOLD is detected as being LOW, the processor lowers HLDA, and when the processor needs to run another cycle, it will again drive the local bus and control lines. HOLD and HLDA have internal pull-up resistors.<br>Hold is not an asynchronous input. External synchronization should be provided if the system cannot otherwise guarantee the set up time. |
| $\overline{SSO}$ | 34 | O | **STATUS LINE:** is logically equivalent to $\overline{S0}$ in the maximum mode. The combination of $\overline{SSO}$, IO/$\overline{M}$ and DT/$\overline{R}$ allows the system to completely decode the current bus cycle status. |

| IO/$\overline{M}$ | DT/$\overline{R}$ | $\overline{SSO}$ | Characteristics |
|---|---|---|---|
| 1(HIGH) | 0 | 0 | Interrupt Acknowledge |
| 1 | 0 | 1 | Read I/O Port |
| 1 | 1 | 0 | Write I/O Port |
| 1 | 1 | 1 | Halt |
| 0(LOW) | 0 | 0 | Code Access |
| 0 | 0 | 1 | Read Memory |
| 0 | 1 | 0 | Write Memory |
| 0 | 1 | 1 | Passive |

*The following pin function descriptions are for the 8088/8288 system in maximum mode (i.e., MN/$\overline{MX}$ = GND). Only the pin functions which are unique to maximum mode are described; all other pin functions are as described above.*

| Symbol | Pin No. | Type | Name and Function |
|---|---|---|---|
| $\overline{S2}$, $\overline{S1}$, $\overline{S0}$ | 26–28 | O | **STATUS:** is active during clock high of T4, T1, and T2, and is returned to the passive state (1,1,1) during T3 or during Tw when READY is HIGH. This status is used by the 8288 bus controller to generate all memory and I/O access control signals. Any change by $\overline{S2}$, $\overline{S1}$, or $\overline{S0}$ during T4 is used to indicate the beginning of a bus cycle, and the return to the passive state in T3 and Tw is used to indicate the end of a bus cycle.<br>These signals float to 3-state OFF during ''hold acknowledge''. During the first clock cycle after RESET becomes active, these signals are active HIGH. After this first clock, they float to 3-state OFF. |

| $\overline{S2}$ | $\overline{S1}$ | $\overline{S0}$ | Characteristics |
|---|---|---|---|
| 0(LOW) | 0 | 0 | Interrupt Acknowledge |
| 0 | 0 | 1 | Read I/O Port |
| 0 | 1 | 0 | Write I/O Port |
| 0 | 1 | 1 | Halt |
| 1(HIGH) | 0 | 0 | Code Access |
| 1 | 0 | 1 | Read Memory |
| 1 | 1 | 0 | Write Memory |
| 1 | 1 | 1 | Passive |

**Table 1. Pin Description** (Continued)

| Symbol | Pin No. | Type | Name and Function |
|---|---|---|---|
| $\overline{RQ}/\overline{GT0}$, $\overline{RQ}/\overline{GT1}$ | 30, 31 | I/O | **REQUEST/GRANT:** pins are used by other local bus masters to force the processor to release the local bus at the end of the processor's current bus cycle. Each pin is bidirectional with $\overline{RQ}/\overline{GT0}$ having higher priority than $\overline{RQ}/\overline{GT1}$. $\overline{RQ}/\overline{GT}$ has an internal pull-up resistor, so may be left unconnected. The request/grant sequence is as follows (See Figure 8):<br>1. A pulse of one CLK wide from another local bus master indicates a local bus request (''hold'') to the 8088 (pulse 1).<br>2. During a T4 or TI clock cycle, a pulse one clock wide from the 8088 to the requesting master (pulse 2), indicates that the 8088 has allowed the local bus to float and that it will enter the ''hold acknowledge'' state at the next CLK. The CPU's bus interface unit is disconnected logically from the local bus during ''hold acknowledge''. The same rules as for HOLD/HOLDA apply as for when the bus is released.<br>3. A pulse one CLK wide from the requesting master indicates to the 8088 (pulse 3) that the ''hold'' request is about to end and that the 8088 can reclaim the local bus at the next CLK. The CPU then enters T4.<br>Each master-master exchange of the local bus is a sequence of three pulses. There must be one idle CLK cycle after each bus exchange. Pulses are active LOW.<br>If the request is made while the CPU is performing a memory cycle, it will release the local bus during T4 of the cycle when all the following conditions are met:<br>1. Request occurs on or before T2.<br>2. Current cycle is not the low bit of a word.<br>3. Current cycle is not the first acknowledge of an interrupt acknowledge sequence.<br>4. A locked instruction is not currently executing.<br>If the local bus is idle when the request is made the two possible events will follow:<br>1. Local bus will be released during the next clock.<br>2. A memory cycle will start within 3 clocks. Now the four rules for a currently active memory cycle apply with condition number 1 already satisfied. |
| $\overline{LOCK}$ | 29 | O | **$\overline{LOCK}$:** indicates that other system bus masters are not to gain control of the system bus while $\overline{LOCK}$ is active (LOW). The $\overline{LOCK}$ signal is activated by the ''LOCK'' prefix instruction and remains active until the completion of the next instruction. This signal is active LOW, and floats to 3-state off in ''hold acknowledge''. |
| QS1, QS0 | 24, 25 | O | **QUEUE STATUS:** provide status to allow external tracking of the internal 8088 instruction queue.<br>The queue status is valid during the CLK cycle after which the queue operation is performed. |

| QS1 | QS0 | Characteristics |
|---|---|---|
| 0(LOW) | 0 | No Operation |
| 0 | 1 | First Byte of Opcode from Queue |
| 1(HIGH) | 0 | Empty the Queue |
| 1 | 1 | Subsequent Byte from Queue |

| | | | |
|---|---|---|---|
| — | 34 | O | Pin 34 is always high in the maximum mode. |

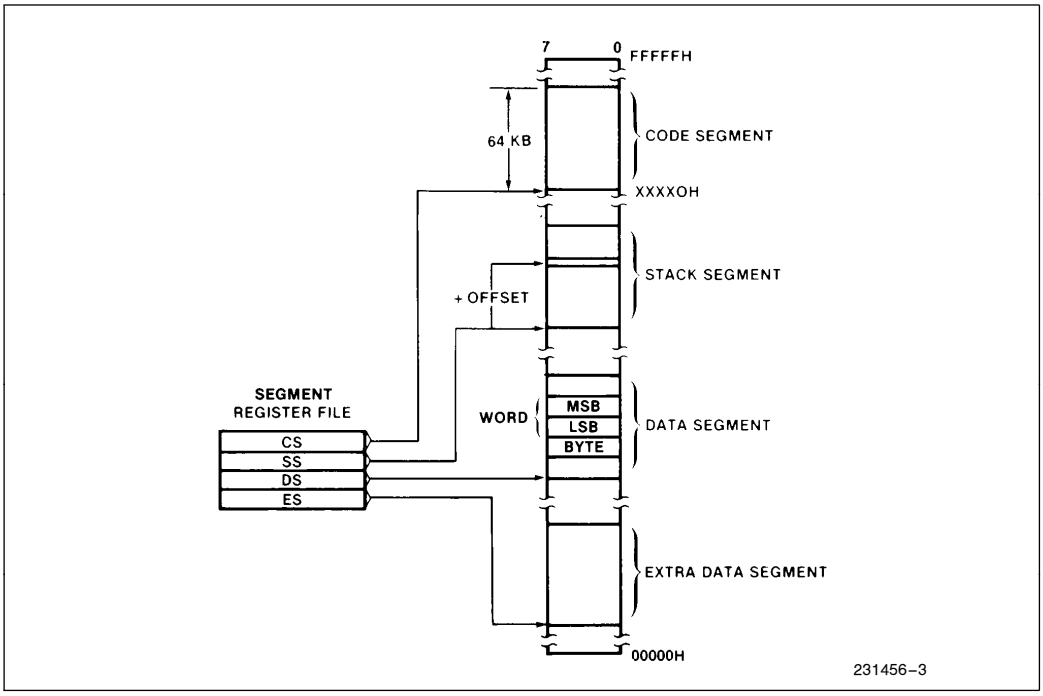



**Figure 3. Memory Organization**

## FUNCTIONAL DESCRIPTION

### Memory Organization

The processor provides a 20-bit address to memory which locates the byte being referenced. The memory is organized as a linear array of up to 1 million bytes, addressed as 00000(H) to FFFFF(H). The memory is logically divided into code, data, extra data, and stack segments of up to 64K bytes each, with each segment falling on 16-byte boundaries (See Figure 3).

All memory references are made relative to base addresses contained in high speed segment registers. The segment types were chosen based on the addressing needs of programs. The segment register to be selected is automatically chosen according to the rules of the following table. All information in one segment type share the same logical attributes (e.g. code or data). By structuring memory into relocatable areas of similar characteristics and by automatically selecting segment registers, programs are shorter, faster, and more structured.

Word (16-bit) operands can be located on even or odd address boundaries. For address and data operands, the least significant byte of the word is stored in the lower valued address location and the most significant byte in the next higher address location. The BIU will automatically execute two fetch or write cycles for 16-bit operands.

| Memory Reference Used | Segment Register Used | Segment Selection Rule |
|---|---|---|
| Instructions | CODE (CS) | Automatic with all instruction prefetch. |
| Stack | STACK (SS) | All stack pushes and pops. Memory references relative to BP base register except data references. |
| Local Data | DATA (DS) | Data references when: relative to stack, destination of string operation, or explicity overridden. |
| External (Global) Data | EXTRA (ES) | Destination of string operations: Explicitly selected using a segment override. |

Certain locations in memory are reserved for specific CPU operations (See Figure 4). Locations from addresses FFFF0H through FFFFFH are reserved for operations including a jump to the initial system initialization routine. Following RESET, the CPU will always begin execution at location FFFF0H where the jump must be located. Locations 00000H through 003FFH are reserved for interrupt operations. Four-byte pointers consisting of a 16-bit segment address and a 16-bit offset address direct program flow to one of the 256 possible interrupt service routines. The pointer elements are assumed to have been stored at their respective places in reserved memory prior to the occurrence of interrupts.

## Minimum and Maximum Modes

The requirements for supporting minimum and maximum 8088 systems are sufficiently different that they cannot be done efficiently with 40 uniquely defined pins. Consequently, the 8088 is equipped with a strap pin (MN/$\overline{MX}$) which defines the system configuration. The definition of a certain subset of the pins changes, dependent on the condition of the strap pin. When the MN/$\overline{MX}$ pin is strapped to GND, the 8088 defines pins 24 through 31 and 34 in maximum mode. When the MN/$\overline{MX}$ pin is strapped to $V_{CC}$, the 8088 generates bus control signals itself on pins 24 through 31 and 34.

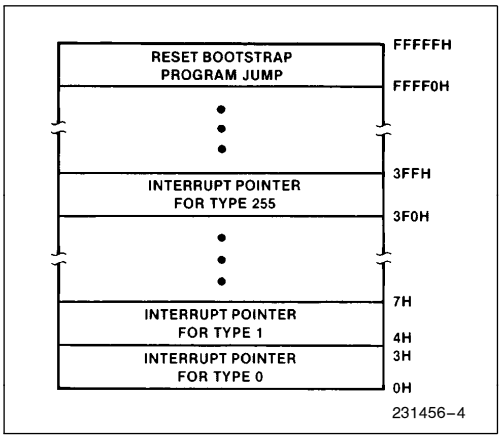



**Figure 4. Reserved Memory Locations**

The minimum mode 8088 can be used with either a multiplexed or demultiplexed bus. The multiplexed bus configuration is compatible with the MCS-85 multiplexed bus peripherals. This configuration (See Figure 5) provides the user with a minimum chip count system. This architecture provides the 8088 processing power in a highly integrated form.

The demultiplexed mode requires one latch (for 64K addressability) or two latches (for a full megabyte of addressing). A third latch can be used for buffering if the address bus loading requires it. A transceiver can also be used if data bus buffering is required (See Figure 6). The 8088 provides $\overline{DEN}$ and DT/$\overline{R}$ to control the transceiver, and ALE to latch the addresses. This configuration of the minimum mode provides the standard demultiplexed bus structure with heavy bus buffering and relaxed bus timing requirements.

The maximum mode employs the 8288 bus controller (See Figure 7). The 8288 decodes status lines $\overline{S0}$, $\overline{S1}$, and $\overline{S2}$, and provides the system with all bus control signals. Moving the bus control to the 8288 provides better source and sink current capability to the control lines, and frees the 8088 pins for extended large system features. Hardware lock, queue status, and two request/grant interfaces are provided by the 8088 in maximum mode. These features allow co-processors in local bus and remote bus configurations.

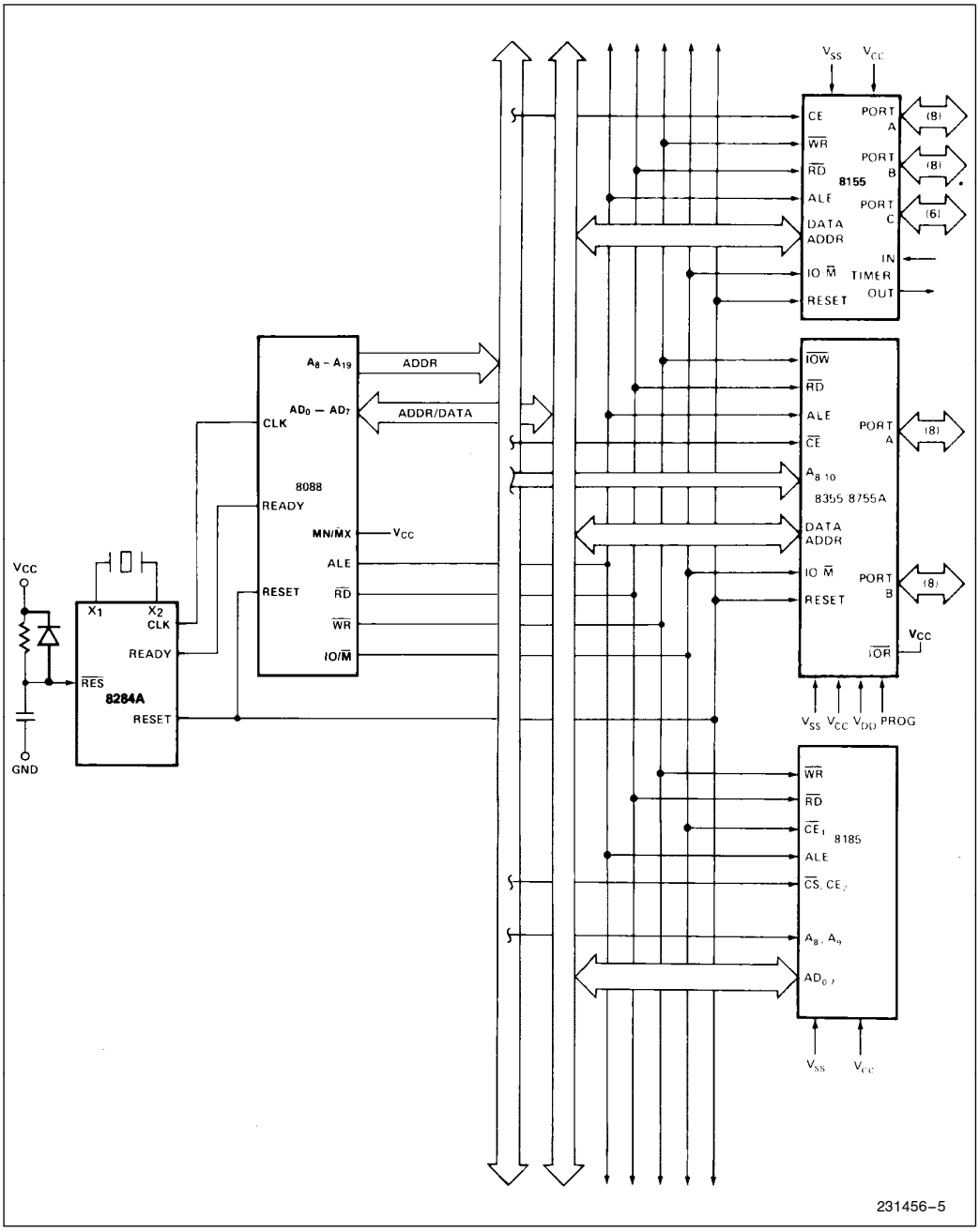



**Figure 5. Multiplexed Bus Configuration**

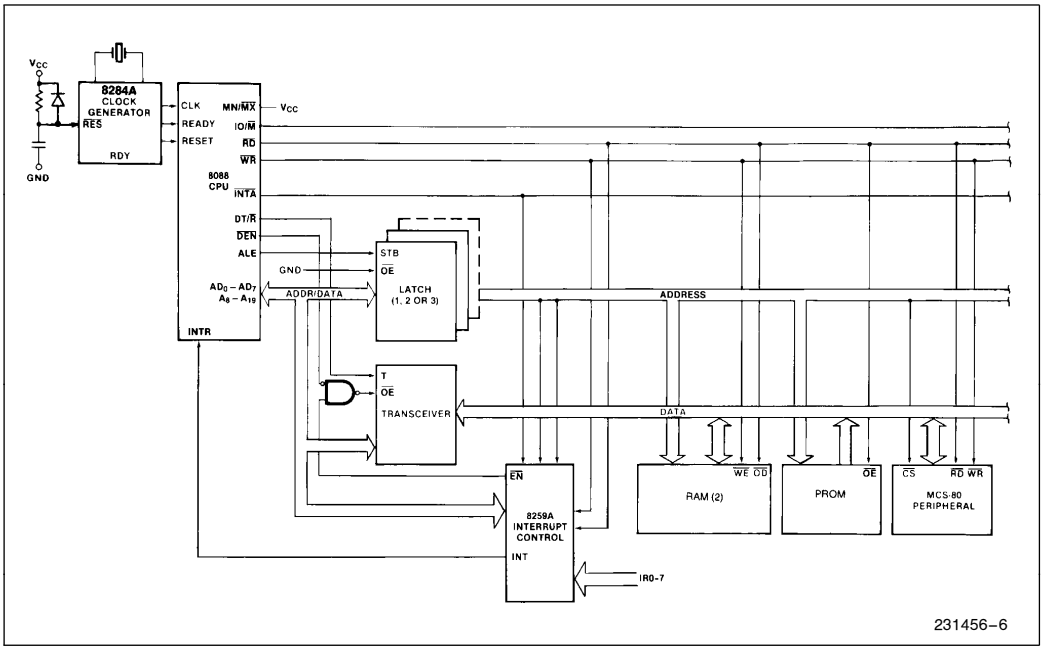



**Figure 6. Demultiplexed Bus Configuration**

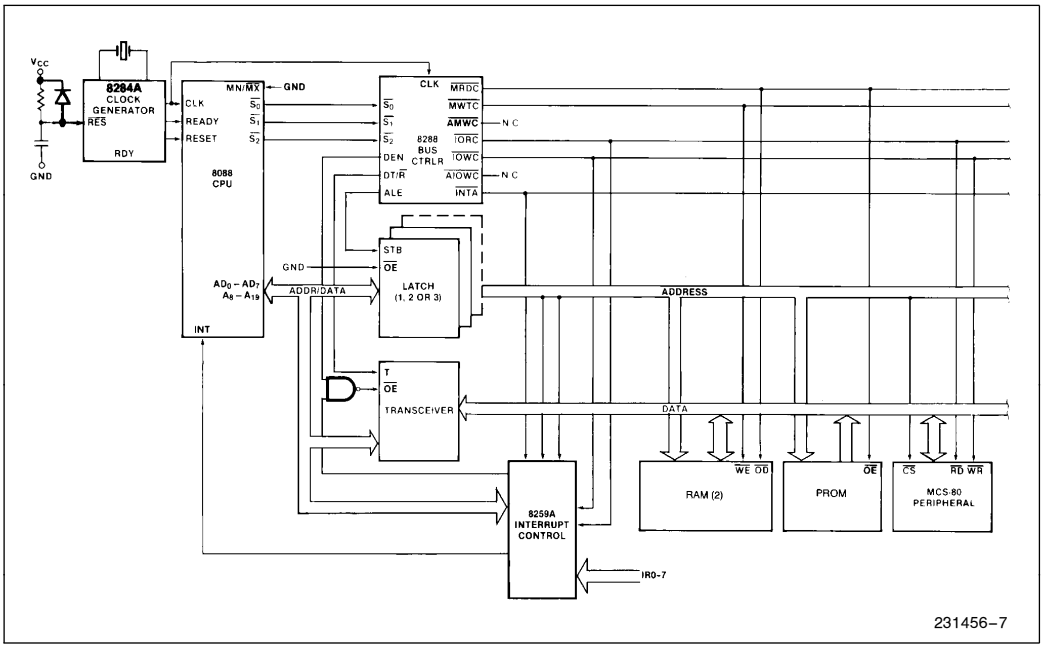



**Figure 7. Fully Buffered System Using Bus Controller**

## Bus Operation

The 8088 address/data bus is broken into three parts—the lower eight address/data bits (AD0–AD7), the middle eight address bits (A8–A15), and the upper four address bits (A16–A19). The address/data bits and the highest four address bits are time multiplexed. This technique provides the most efficient use of pins on the processor, permitting the use of a standard 40 lead package. The middle eight address bits are not multiplexed, i.e. they remain valid throughout each bus cycle. In addition, the bus can be demultiplexed at the processor with a single address latch if a standard, non-multiplexed bus is desired for the system.

Each processor bus cycle consists of at least four CLK cycles. These are referred to as T1, T2, T3, and T4 (See Figure 8). The address is emitted from the processor during T1 and data transfer occurs on the bus during T3 and T4. T2 is used primarily for chang-

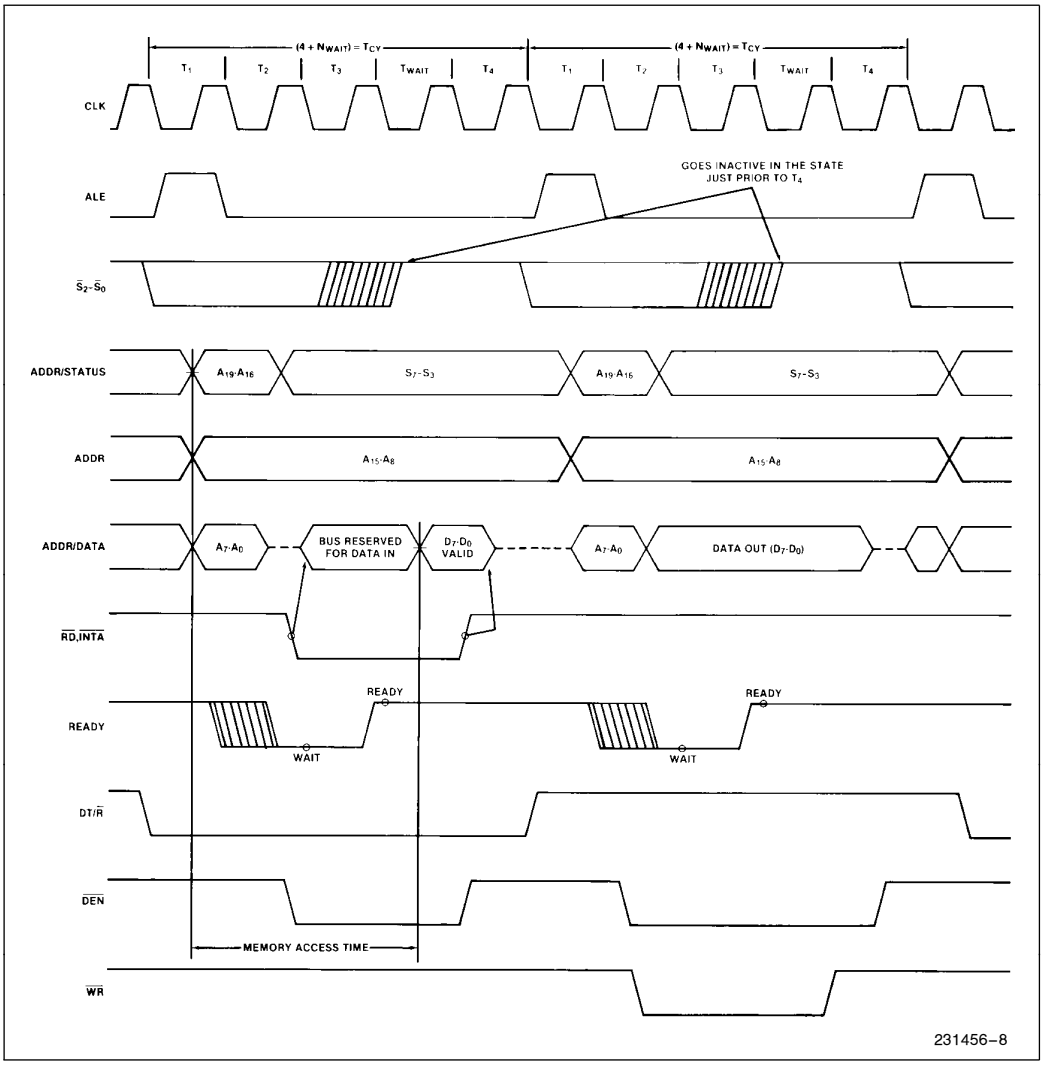



**Figure 8. Basic System Timing**

ing the direction of the bus during read operations. In the event that a "NOT READY" indication is given by the addressed device, "wait" states (Tw) are inserted between T3 and T4. Each inserted "wait" state is of the same duration as a CLK cycle. Periods can occur between 8088 driven bus cycles. These are referred to as "idle" states (Ti), or inactive CLK cycles. The processor uses these cycles for internal housekeeping.

During T1 of any bus cycle, the ALE (address latch enable) signal is emitted (by either the processor or the 8288 bus controller, depending on the MN/$\overline{MX}$ strap). At the trailing edge of this pulse, a valid address and certain status information for the cycle may be latched.

Status bits $\overline{S0}$, $\overline{S1}$, and $\overline{S2}$ are used by the bus controller, in maximum mode, to identify the type of bus transaction according to the following table:

| $\overline{S2}$ | $\overline{S1}$ | $\overline{S0}$ | Characteristics |
|---|---|---|---|
| 0(LOW) | 0 | 0 | Interrupt Acknowledge |
| 0 | 0 | 1 | Read I/O |
| 0 | 1 | 0 | Write I/O |
| 0 | 1 | 1 | Halt |
| 1(HIGH) | 0 | 0 | Instruction Fetch |
| 1 | 0 | 1 | Read Data from Memory |
| 1 | 1 | 0 | Write Data to Memory |
| 1 | 1 | 1 | Passive (No Bus Cycle) |

Status bits S3 through S6 are multiplexed with high order address bits and are therefore valid during T2 through T4. S3 and S4 indicate which segment register was used for this bus cycle in forming the address according to the following table:

| $S_4$ | $S_3$ | Characteristics |
|---|---|---|
| 0(LOW) | 0 | Alternate Data (Extra Segment) |
| 0 | 1 | Stack |
| 1(HIGH) | 0 | Code or None |
| 1 | 1 | Data |

S5 is a reflection of the PSW interrupt enable bit. S6 is always equal to 0.

## I/O Addressing

In the 8088, I/O operations can address up to a maximum of 64K I/O registers. The I/O address appears in the same format as the memory address on bus lines A15–A0. The address lines A19–A16 are zero in I/O operations. The variable I/O instructions, which use register DX as a pointer, have full address capability, while the direct I/O instructions directly address one or two of the 256 I/O byte locations in page 0 of the I/O address space. I/O ports are addressed in the same manner as memory locations.

Designers familiar with the 8085 or upgrading an 8085 design should note that the 8085 addresses I/O with an 8-bit address on both halves of the 16-bit address bus. The 8088 uses a full 16-bit address on its lower 16 address lines.

# EXTERNAL INTERFACE

## Processor Reset and Initialization

Processor initialization or start up is accomplished with activation (HIGH) of the RESET pin. The 8088 RESET is required to be HIGH for greater than four clock cycles. The 8088 will terminate operations on the high-going edge of RESET and will remain dormant as long as RESET is HIGH. The low-going transition of RESET triggers an internal reset sequence for approximately 7 clock cycles. After this interval the 8088 operates normally, beginning with the instruction in absolute locations FFFF0H (See Figure 4). The RESET input is internally synchronized to the processor clock. At initialization, the HIGH to LOW transition of RESET must occur no sooner than 50 $\mu$s after power up, to allow complete initialization of the 8088.

NMI asserted prior to the 2nd clock after the end of RESET will not be honored. If NMI is asserted after that point and during the internal reset sequence, the processor may execute one instruction before responding to the interrupt. A hold request active immediately after RESET will be honored before the first instruction fetch.

All 3-state outputs float to 3-state OFF during RESET. Status is active in the idle state for the first clock after RESET becomes active and then floats to 3-state OFF. ALE and HLDA are driven low.

## Interrupt Operations

Interrupt operations fall into two classes: software or hardware initiated. The software initiated interrupts and software aspects of hardware interrupts are specified in the instruction set description in the iAPX 88 book or the iAPX 86,88 User's Manual. Hardware interrupts can be classified as nonmaskable or maskable.

Interrupts result in a transfer of control to a new program location. A 256 element table containing address pointers to the interrupt service program locations resides in absolute locations 0 through 3FFH (See Figure 4), which are reserved for this purpose. Each element in the table is 4 bytes in size and corresponds to an interrupt ''type.'' An interrupting device supplies an 8-bit type number, during the interrupt acknowledge sequence, which is used to vector through the appropriate element to the new interrupt service program location.

## Non-Maskable Interrupt (NMI)

The processor provides a single non-maskable interrupt (NMI) pin which has higher priority than the maskable interrupt request (INTR) pin. A typical use would be to activate a power failure routine. The NMI is edge-triggered on a LOW to HIGH transition. The activation of this pin causes a type 2 interrupt.

NMI is required to have a duration in the HIGH state of greater than two clock cycles, but is not required to be synchronized to the clock. Any higher going transition of NMI is latched on-chip and will be serviced at the end of the current instruction or between whole moves (2 bytes in the case of word moves) of a block type instruction. Worst case response to NMI would be for multiply, divide, and variable shift instructions. There is no specification on the occurrence of the low-going edge; it may occur before, during, or after the servicing of NMI. Another high-going edge triggers another response if it occurs after the start of the NMI procedure. The signal must be free of logical spikes in general and be free of bounces on the low-going edge to avoid triggering extraneous responses.

## Maskable Interrupt (INTR)

The 8088 provides a single interrupt request input (INTR) which can be masked internally by software with the resetting of the interrupt enable (IF) flag bit. The interrupt request signal is level triggered. It is internally synchronized during each clock cycle on the high-going edge of CLK. To be responded to, INTR must be present (HIGH) during the clock period preceding the end of the current instruction or the end of a whole move for a block type instruction. During interrupt response sequence, further interrupts are disabled. The enable bit is reset as part of the response to any interrupt (INTR, NMI, software interrupt, or single step), although the FLAGS register which is automatically pushed onto the stack reflects the state of the processor prior to the interrupt. Until the old FLAGS register is restored, the enable bit will be zero unless specifically set by an instruction.

During the response sequence (See Figure 9), the processor executes two successive (back to back) interrupt acknowledge cycles. The 8088 emits the LOCK signal (maximum mode only) from T2 of the first bus cycle until T2 of the second. A local bus ''hold'' request will not be honored until the end of the second bus cycle. In the second bus cycle, a byte is fetched from the external interrupt system (e.g., 8259A PIC) which identifies the source (type) of the interrupt. This byte is multiplied by four and used as a pointer into the interrupt vector lookup table. An INTR signal left HIGH will be continually responded to within the limitations of the enable bit and sample period. The interrupt return instruction includes a flags pop which returns the status of the original interrupt enable bit when it restores the flags.

## HALT

When a software HALT instruction is executed, the processor indicates that it is entering the HALT state in one of two ways, depending upon which mode is strapped. In minimum mode, the processor issues ALE, delayed by one clock cycle, to allow the system to latch the halt status. Halt status is available on IO/$\overline{M}$, DT/$\overline{R}$, and $\overline{SSO}$. In maximum mode, the processor issues appropriate HALT status on $\overline{S2}$, $\overline{S1}$, and $\overline{S0}$, and the 8288 bus controller issues one ALE. The 8088 will not leave the HALT state when a local bus hold is entered while in HALT. In this case, the processor reissues the HALT indicator at the end of the local bus hold. An interrupt request or RESET will force the 8088 out of the HALT state.

## Read/Modify/Write (Semaphore) Operations via LOCK

The LOCK status information is provided by the processor when consecutive bus cycles are required during the execution of an instruction. This allows the processor to perform read/modify/write operations on memory (via the ''exchange register with memory'' instruction), without another system bus master receiving intervening memory cycles. This is useful in multiprocessor system configurations to accomplish ''test and set lock'' operations. The $\overline{LOCK}$ signal is activated (LOW) in the clock cycle following decoding of the LOCK prefix instruction. It is deactivated at the end of the last bus cycle of the instruction following the LOCK prefix. While $\overline{LOCK}$ is active, a request on a $\overline{RQ}$/$\overline{GT}$ pin will be recorded, and then honored at the end of the LOCK.

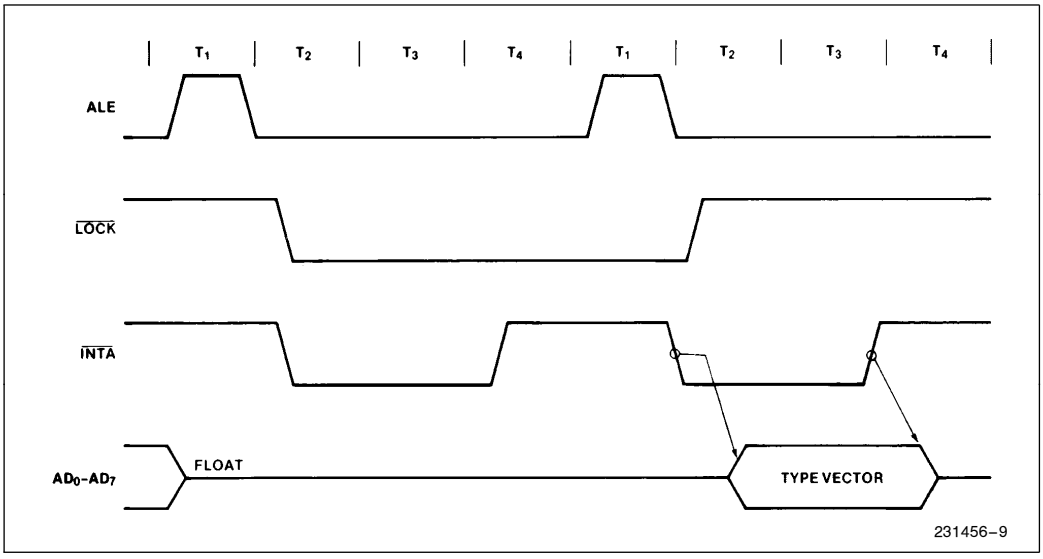



**Figure 9. Interrupt Acknowledge Sequence**

## External Synchronization via TEST

As an alternative to interrupts, the 8088 provides a single software-testable input pin (TEST). This input is utilized by executing a WAIT instruction. The single WAIT instruction is repeatedly executed until the TEST input goes active (LOW). The execution of WAIT does not consume bus cycles once the queue is full.

If a local bus request occurs during WAIT execution, the 8088 3-states all output drivers. If interrupts are enabled, the 8088 will recognize interrupts and process them. The WAIT instruction is then refetched, and reexecuted.

## Basic System Timing

In minimum mode, the MN/MX pin is strapped to $V_{CC}$ and the processor emits bus control signals compatible with the 8085 bus structure. In maximum mode, the MN/MX pin is strapped to GND and the processor emits coded status information which the 8288 bus controller uses to generate MULTIBUS compatible bus control signals.

## System Timing—Minimum System

(See Figure 8)

The read cycle begins in T1 with the assertion of the address latch enable (ALE) signal. The trailing (low going) edge of this signal is used to latch the address information, which is valid on the address/data bus (AD0–AD7) at this time, into the 8282/8283 latch. Address lines A8 through A15 do not need to be latched because they remain valid throughout the bus cycle. From T1 to T4 the IO/M signal indicates a memory or I/O operation. At T2 the address is removed from the address/data bus and the bus goes to a high impedance state. The read control signal is also asserted at T2. The read (RD) signal causes the addressed device to enable its data bus drivers to the local bus. Some time later, valid data will be available on the bus and the addressed device will drive the READY line HIGH. When the processor returns the read signal to a HIGH level, the addressed device will again 3-state its bus drivers. If a transceiver is required to buffer the 8088 local bus, signals DT/R and DEN are provided by the 8088.

A write cycle also begins with the assertion of ALE and the emission of the address. The IO/M signal is again asserted to indicate a memory or I/O write operation. In T2, immediately following the address emission, the processor emits the data to be written into the addressed location. This data remains valid until at least the middle of T4. During T2, T3, and Tw, the processor asserts the write control signal. The write (WR) signal becomes active at the beginning of T2, as opposed to the read, which is delayed somewhat into T2 to provide time for the bus to float.

The basic difference between the interrupt acknowledge cycle and a read cycle is that the interrupt acknowledge ($\overline{INTA}$) signal is asserted in place of the read ($\overline{RD}$) signal and the address bus is floated. (See Figure 9) In the second of two successive $\overline{INTA}$ cycles, a byte of information is read from the data bus, as supplied by the interrupt system logic (i.e. 8259A priority interrupt controller). This byte identifies the source (type) of the interrupt. It is multiplied by four and used as a pointer into the interrupt vector lookup table, as described earlier.

## Bus Timing—Medium Complexity Systems

(See Figure 10)

For medium complexity systems, the MN/$\overline{MX}$ pin is connected to GND and the 8288 bus controller is added to the system, as well as a latch for latching the system address, and a transceiver to allow for bus loading greater than the 8088 is capable of handling. Signals ALE, $\overline{DEN}$, and DT/$\overline{R}$ are generated by the 8288 instead of the processor in this configuration, although their timing remains relatively the same. The 8088 status outputs ($\overline{S2}$, $\overline{S1}$, and $\overline{S0}$) provide type of cycle information and become 8288 inputs. This bus cycle information specifies read (code, data, or I/O), write (data or I/O), interrupt acknowledge, or software halt. The 8288 thus issues control signals specifying memory read or write, I/O read or write, or interrupt acknowledge. The 8288 provides two types of write strobes, normal and advanced, to be applied as required. The normal write strobes have data valid at the leading edge of write. The advanced write strobes have the same timing as read strobes, and hence, data is not valid at the leading edge of write. The transceiver receives the usual T and $\overline{OE}$ inputs from the 8288's DT/$\overline{R}$ and $\overline{DEN}$ outputs.

The pointer into the interrupt vector table, which is passed during the second $\overline{INTA}$ cycle, can derive from an 8259A located on either the local bus or the system bus. If the master 8289A priority interrupt controller is positioned on the local bus, a TTL gate is required to disable the transceiver when reading from the master 8259A during the interrupt acknowledge sequence and software ''poll''.

## The 8088 Compared to the 8086

The 8088 CPU is an 8-bit processor designed around the 8086 internal structure. Most internal functions of the 8088 are identical to the equivalent 8086 functions. The 8088 handles the external bus the same way the 8086 does with the distinction of handling only 8 bits at a time. Sixteen-bit operands are fetched or written in two consecutive bus cycles. Both processors will appear identical to the software engineer, with the exception of execution time. The internal register structure is identical and all instructions have the same end result. The differences between the 8088 and 8086 are outlined below. The engineer who is unfamiliar with the 8086 is referred to the iAPX 86, 88 User's Manual, Chapters 2 and 4, for function description and instruction set information. Internally, there are three differences between the 8088 and the 8086. All changes are related to the 8-bit bus interface.

- The queue length is 4 bytes in the 8088, whereas the 8086 queue contains 6 bytes, or three words. The queue was shortened to prevent overuse of the bus by the BIU when prefetching instructions. This was required because of the additional time necessary to fetch instructions 8 bits at a time.
- To further optimize the queue, the prefetching algorithm was changed. The 8088 BIU will fetch a new instruction to load into the queue each time there is a 1 byte hole (space available) in the queue. The 8086 waits until a 2-byte space is available.
- The internal execution time of the instruction set is affected by the 8-bit interface. All 16-bit fetches and writes from/to memory take an additional four clock cycles. The CPU is also limited by the speed of instruction fetches. This latter problem only occurs when a series of simple operations occur. When the more sophisticated instructions of the 8088 are being used, the queue has time to fill and the execution proceeds as fast as the execution unit will allow.

The 8088 and 8086 are completely software compatible by virtue of their identical execution units. Software that is system dependent may not be completely transferable, but software that is not system dependent will operate equally as well on an 8088 and an 8086.

The hardware interface of the 8088 contains the major differences between the two CPUs. The pin assignments are nearly identical, however, with the following functional changes:

- A8–A15—These pins are only address outputs on the 8088. These address lines are latched internally and remain valid throughout a bus cycle in a manner similar to the 8085 upper address lines.
- $\overline{BHE}$ has no meaning on the 8088 and has been eliminated.

- $\overline{SSO}$ provides the $\overline{SO}$ status information in the minimum mode. This output occurs on pin 34 in minimum mode only. DT/$\overline{R}$, IO/$\overline{M}$, and $\overline{SSO}$ provide the complete bus status in minimum mode.
- IO/$\overline{M}$ has been inverted to be compatible with the MCS-85 bus structure.
- ALE is delayed by one clock cycle in the minimum mode when entering HALT, to allow the status to be latched with ALE.

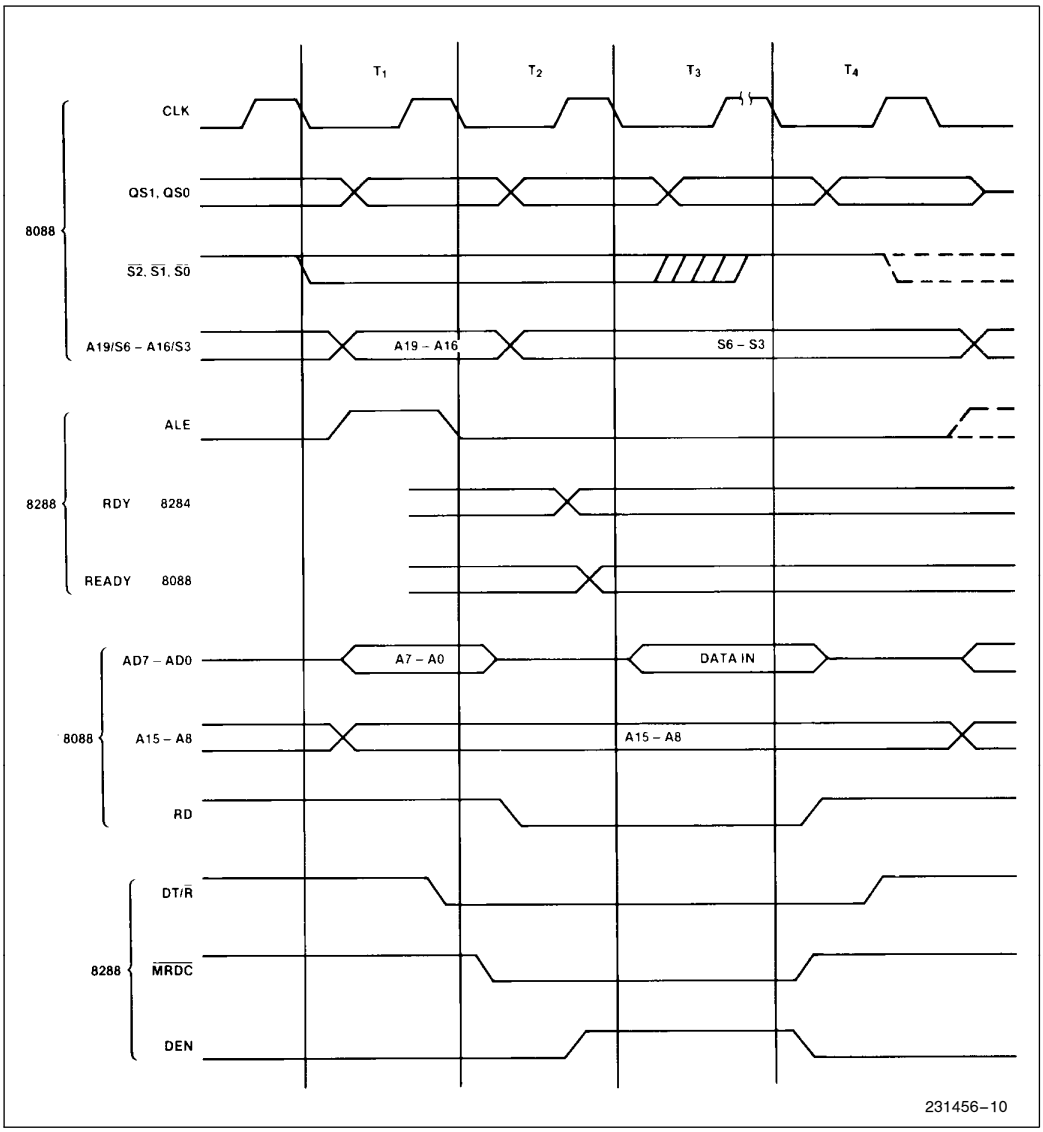



**Figure 10. Medium Complexity System Timing**

## ABSOLUTE MAXIMUM RATINGS*

Ambient Temperature Under Bias . . . . 0°C to +70°C
Case Temperature (Plastic) . . . . . 0°C to +95°C
Case Temperature (CERDIP) . . . . . 0°C to +75°C
Storage Temperature . . . . . −65°C to +150°C
Voltage on Any Pin with
Respect to Ground . . . . . −1.0 to +7V
Power Dissipation . . . . . 2.5 Watt

NOTICE: This is a production data sheet. The specifications are subject to change without notice.

**WARNING: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.*

## D.C. CHARACTERISTICS

($T_A$ = 0°C to 70°C, $T_{CASE}$ (Plastic) = 0°C to 95°C, $T_{CASE}$ (CERDIP) = 0°C to 75°C,
$T_A$ = 0°C to 55°C and $T_{CASE}$ = 0°C to 75°C for P8088-2 only
$T_A$ is guaranteed as long as $T_{CASE}$ is not exceeded)

($V_{CC}$ = 5V ±10% for 8088, $V_{CC}$ = 5V ±5% for 8088-2 and Extended Temperature EXPRESS)

| Symbol | Parameter | Min | Max | Units | Test Conditions |
|---|---|---|---|---|---|
| $V_{IL}$ | Input Low Voltage | −0.5 | +0.8 | V | (Note 1) |
| $V_{IH}$ | Input High Voltage | 2.0 | $V_{CC}$ + 0.5 | V | (Notes 1, 2) |
| $V_{OL}$ | Output Low Voltage | | 0.45 | V | $I_{OL}$ = 2.0 mA |
| $V_{OH}$ | Output High Voltage | 2.4 | | V | $I_{OH}$ = −400 μA |
| $I_{CC}$ | Power Supply Current: 8088<br>8088-2<br>P8088 | | 340<br>350<br>250 | mA | $T_A$ = 25°C |
| $I_{LI}$ | Input Leakage Current | | ±10 | μA | 0V ≤ $V_{IN}$ ≤ $V_{CC}$ (Note 3) |
| $I_{LO}$ | Output and I/O Leakage Current | | ±10 | μA | 0.45V ≤ $V_{OUT}$ ≤ $V_{CC}$ |
| $V_{CL}$ | Clock Input Low Voltage | −0.5 | +0.6 | V | |
| $V_{CH}$ | Clock Input High Voltage | 3.9 | $V_{CC}$ + 1.0 | V | |
| $C_{IN}$ | Capacitance If Input Buffer (All Input Except $AD_0$–$AD_7$, RQ/GT) | | 15 | pF | fc = 1 MHz |
| $C_{IO}$ | Capacitance of I/O Buffer $AD_0$–$AD_7$, RQ/GT) | | 15 | pF | fc = 1 MHz |

**NOTES:**
1. $V_{IL}$ tested with MN/$\overline{MX}$ Pin = 0V
   $V_{IH}$ tested with MN/$\overline{MX}$ Pin = 5V
   MN/$\overline{MX}$ Pin is a strap Pin
2. Not applicable to $\overline{RQ}$/$\overline{GT0}$ and $\overline{RQ}$/$\overline{GT1}$ Pins (Pins 30 and 31)
3. HOLD and HLDA $I_{LI}$ Min = 30 μA, Max = 500 μA

## A.C. CHARACTERISTICS

($T_A$ = 0°C to 70°C, $T_{CASE}$ (Plastic) = 0°C to 95°C, $T_{CASE}$ (CERDIP) = 0°C to 75°C,
$T_A$ = 0°C to 55°C and $T_{CASE}$ = 0°C to 80°C for P8088-2 only
$T_A$ is guaranteed as long as $T_{CASE}$ is not exceeded)

($V_{CC}$ = 5V ±10% for 8088, $V_{CC}$ = 5V ±5% for 8088-2 and Extended Temperature EXPRESS)

**MINIMUM COMPLEXITY SYSTEM TIMING REQUIREMENTS**

| Symbol | Parameter | 8088 | | 8088-2 | | Units | Test Conditions |
|---|---|---|---|---|---|---|---|
| | | Min | Max | Min | Max | | |
| TCLCL | CLK Cycle Period | 200 | 500 | 125 | 500 | ns | |
| TCLCH | CLK Low Time | 118 | | 68 | | ns | |
| TCHCL | CLK High Time | 69 | | 44 | | ns | |
| TCH1CH2 | CLK Rise Time | | 10 | | 10 | ns | From 1.0V to 3.5V |
| TCL2CL2 | CLK Fall Time | | 10 | | 10 | ns | From 3.5V to 1.0V |
| TDVCL | Data in Setup Time | 30 | | 20 | | ns | |
| TCLDX | Data in Hold Time | 10 | | 10 | | ns | |
| TR1VCL | RDY Setup Time into 8284 (Notes 1, 2) | 35 | | 35 | | ns | |
| TCLR1X | RDY Hold Time into 8284 (Notes 1, 2) | 0 | | 0 | | ns | |
| TRYHCH | READY Setup Time into 8088 | 118 | | 68 | | ns | |
| TCHRYX | READY Hold Time into 8088 | 30 | | 20 | | ns | |
| TRYLCL | READY Inactive to CLK (Note 3) | −8 | | −8 | | ns | |
| THVCH | HOLD Setup Time | 35 | | 20 | | ns | |
| TINVCH | INTR, NMI, $\overline{\text{TEST}}$ Setup Time (Note 2) | 30 | | 15 | | ns | |
| TILIH | Input Rise Time (Except CLK) | | 20 | | 20 | ns | From 0.8V to 2.0V |
| TIHIL | Input Fall Time (Except CLK) | | 12 | | 12 | ns | From 2.0V to 0.8V |

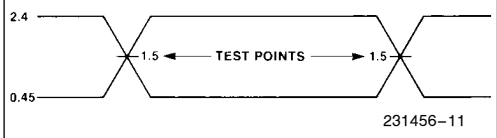
## A.C. CHARACTERISTICS (Continued)

**TIMING RESPONSES**

| Symbol | Parameter | 8088 | | 8088-2 | | Units | Test Conditions |
|---|---|---|---|---|---|---|---|
| | | Min | Max | Min | Max | | |
| TCLAV | Address Valid Delay | 10 | 110 | 10 | 60 | ns | |
| TCLAX | Address Hold Time | 10 | | 10 | | ns | |
| TCLAZ | Address Float Delay | TCLAX | 80 | TCLAX | 50 | ns | |
| TLHLL | ALE Width | TCLCH − 20 | | TCLCH − 10 | | ns | |
| TCLLH | ALE Active Delay | | 80 | | 50 | ns | |
| TCHLL | ALE Inactive Delay | | 85 | | 55 | ns | |
| TLLAX | Address Hold Time to ALE Inactive | TCHCL − 10 | | TCHCL − 10 | | ns | |
| TCLDV | Data Valid Delay | 10 | 110 | 10 | 60 | ns | |
| TCHDX | Data Hold Time | 10 | | 10 | | ns | |
| TWHDX | Data Hold Time after $\overline{WR}$ | TCLCH − 30 | | TCLCH − 30 | | ns | |
| TCVCTV | Control Active Delay 1 | 10 | 110 | 10 | 70 | ns | |
| TCHCTV | Control Active Delay 2 | 10 | 110 | 10 | 60 | ns | |
| TCVCTX | Control Inactive Delay | 10 | 110 | 10 | 70 | ns | |
| TAZRL | Address Float to READ Active | 0 | | 0 | | ns | |
| TCLRL | $\overline{RD}$ Active Delay | 10 | 165 | 10 | 100 | ns | |
| TCLRH | $\overline{RD}$ Inactive Delay | 10 | 150 | 10 | 80 | ns | |
| TRHAV | $\overline{RD}$ Inactive to Next Address Active | TCLCL − 45 | | TCLCL − 40 | | ns | |
| TCLHAV | HLDA Valid Delay | 10 | 160 | 10 | 100 | ns | |
| TRLRH | $\overline{RD}$ Width | 2TCLCL − 75 | | 2TCLCL − 50 | | ns | |
| TWLWH | $\overline{WR}$ Width | 2TCLCL − 60 | | 2TCLCL − 40 | | ns | |
| TAVAL | Address Valid to ALE Low | TCLCH − 60 | | TCLCH − 40 | | ns | |
| TOLOH | Output Rise Time | | 20 | | 20 | ns | From 0.8V to 2.0V |
| TOHOL | Output Fall Time | | 12 | | 12 | ns | From 2.0V to 0.8V |

**NOTES:**
1. Signal at 8284A shown for reference only. See 8284A data sheet for the most recent specifications.
2. Set up requirement for asynchronous signal only to guarantee recognition at next CLK.
3. Applies only to T2 state (8 ns into T3 state).

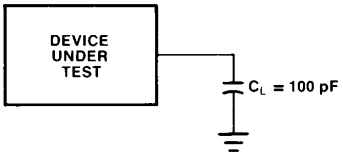
### A.C. TESTING INPUT, OUTPUT WAVEFORM




231456–11

A.C. Testing; Inputs are driven at 2.4V for a logic "1" and 0.45V for a logic "0". Timing measurements are made at 1.5V for both a logic "1" and logic "0".

### A.C. TESTING LOAD CIRCUIT

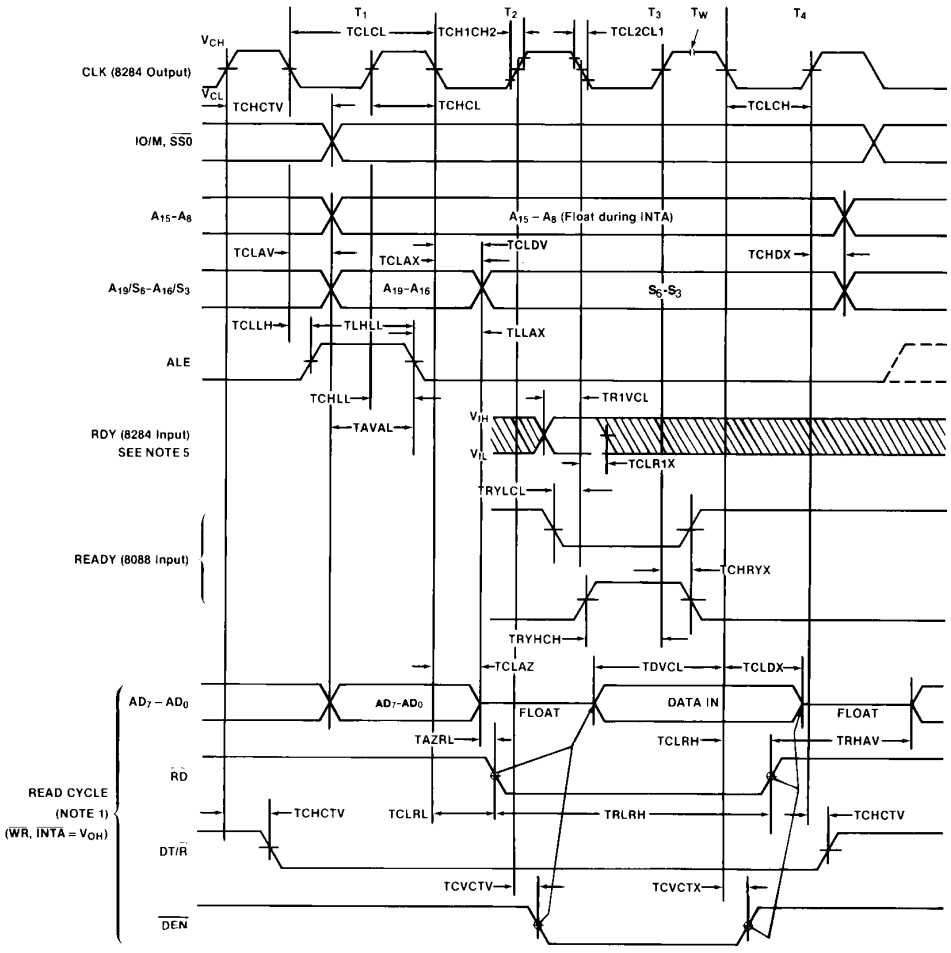



231456–12

$C_L$ Includes Jig Capacitance

## WAVEFORMS

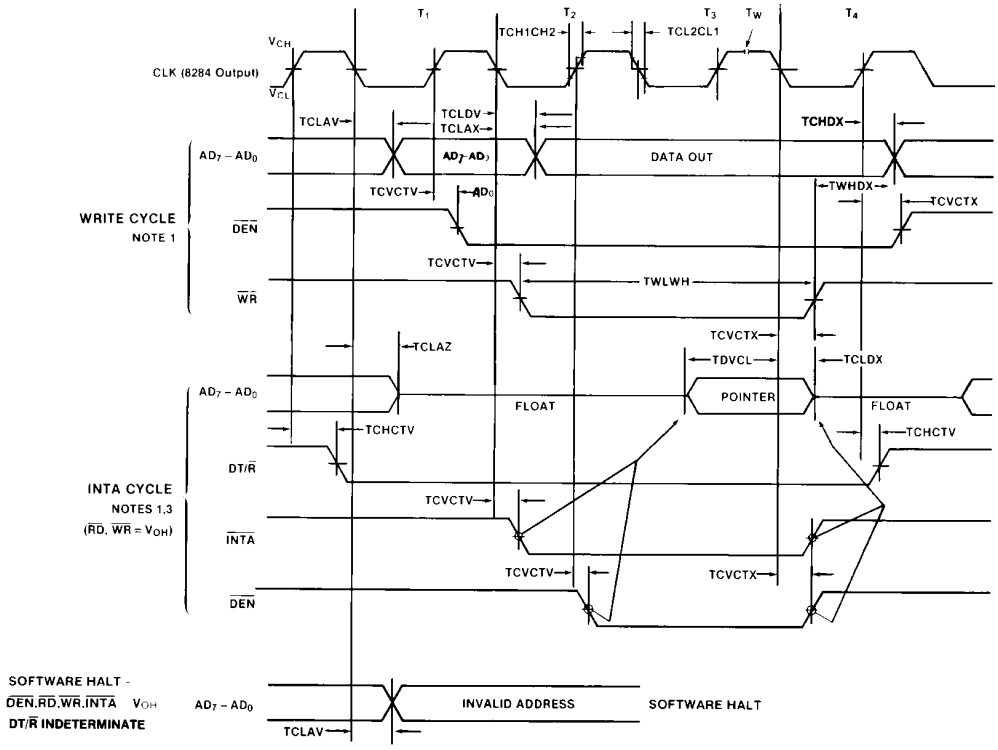
### BUS TIMING—MINIMUM MODE SYSTEM




231456–13

## WAVEFORMS (Continued)

### BUS TIMING—MINIMUM MODE SYSTEM (Continued)




231456–14

**NOTES:**
1. All signals switch between $V_{OH}$ and $V_{OL}$ unless otherwise specified.
2. RDY is sampled near the end of $T_2$, $T_3$, $T_W$ to determine if $T_W$ machines states are to be inserted.
3. Two INTA cycles run back-to-back. The 8088 local ADDR/DATA bus is floating during both INTA cycles. Control signals are shown for the second INTA cycle.
4. Signals at 8284 are shown for reference only.
5. All timing measurements are made at 1.5V unless otherwise noted.

## A.C. CHARACTERISTICS

### MAX MODE SYSTEM (USING 8288 BUS CONTROLLER)

**TIMING REQUIREMENTS**

| Symbol | Parameter | 8088 | | 8088-2 | | Units | Test Conditions |
|---|---|---|---|---|---|---|---|
| | | Min | Max | Min | Max | | |
| TCLCL | CLK Cycle Period | 200 | 500 | 125 | 500 | ns | |
| TCLCH | CLK Low Time | 118 | | 68 | | ns | |
| TCHCL | CLK High Time | 69 | | 44 | | ns | |
| TCH1CH2 | CLK Rise Time | | 10 | | 10 | ns | From 1.0V to 3.5V |
| TCL2CL1 | CLK Fall Time | | 10 | | 10 | ns | From 3.5V to 1.0V |
| TDVCL | Data in Setup Time | 30 | | 20 | | ns | |
| TCLDX | Data in Hold Time | 10 | | 10 | | ns | |
| TR1VCL | RDY Setup Time into 8284 (Notes 1, 2) | 35 | | 35 | | ns | |
| TCLR1X | RDY Hold Time into 8284 (Notes 1, 2) | 0 | | 0 | | ns | |
| TRYHCH | READY Setup Time into 8088 | 118 | | 68 | | ns | |
| TCHRYX | READY Hold Time into 8088 | 30 | | 20 | | ns | |
| TRYLCL | READY Inactive to CLK (Note 4) | −8 | | −8 | | ns | |
| TINVCH | Setup Time for Recognition (INTR, NMI, TEST) (Note 2) | 30 | | 15 | | ns | |
| TGVCH | RQ/GT Setup Time | 30 | | 15 | | ns | |
| TCHGX | RQ Hold Time into 8088 | 40 | | 30 | | ns | |
| TILIH | Input Rise Time (Except CLK) | | 20 | | 20 | ns | From 0.8V to 2.0V |
| TIHIL | Input Fall Time (Except CLK) | | 12 | | 12 | ns | From 2.0V to 0.8V |

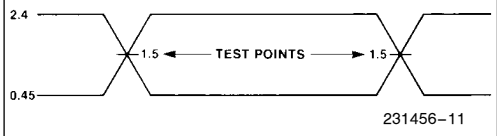
## A.C. CHARACTERISTICS (Continued)

**TIMING RESPONSES**

| Symbol | Parameter | 8088 | | 8088-2 | | Units | Test Conditions |
|---|---|---|---|---|---|---|---|
| | | Min | Max | Min | Max | | |
| TCLML | Command Active Delay (Note 1) | 10 | 35 | 10 | 35 | ns | |
| TCLMH | Command Inactive Delay (Note 1) | 10 | 35 | 10 | 35 | ns | |
| TRYHSH | READY Active to Status Passive (Note 3) | | 110 | | 65 | ns | |
| TCHSV | Status Active Delay | 10 | 110 | 10 | 60 | ns | |
| TCLSH | Status Inactive Delay | 10 | 130 | 10 | 70 | ns | |
| TCLAV | Address Valid Delay | 10 | 110 | 10 | 60 | ns | |
| TCLAX | Address Hold Time | 10 | | 10 | | ns | $C_L$ = 20–100 pF for All 8088 Outputs in Addition to Internal Loads |
| TCLAZ | Address Float Delay | TCLAX | 80 | TCLAX | 50 | ns | |
| TSVLH | Status Valid to ALE High (Note 1) | | 15 | | 15 | ns | |
| TSVMCH | Status Valid to MCE High (Note 1) | | 15 | | 15 | ns | |
| TCLLH | CLK Low to ALE Valid (Note 1) | | 15 | | 15 | ns | |
| TCLMCH | CLK Low to MCE (Note 1) | | 15 | | 15 | ns | |
| TCHLL | ALE Inactive Delay (Note 1) | | 15 | | 15 | ns | |
| TCLMCL | MCE Inactive Delay (Note 1) | | 15 | | 15 | ns | |
| TCLDV | Data Valid Delay | 10 | 110 | 10 | 60 | ns | |
| TCHDX | Data Hold Time | 10 | | 10 | | ns | |
| TCVNV | Control Active Delay (Note 1) | 5 | 45 | 5 | 45 | ns | |
| TCVNX | Control Inactive Delay (Note 1) | 10 | 45 | 10 | 45 | ns | |
| TAZRL | Address Float to Read Active | 0 | | 0 | | ns | |
| TCLRL | $\overline{RD}$ Active Delay | 10 | 165 | 10 | 100 | ns | |
| TCLRH | $\overline{RD}$ Inactive Delay | 10 | 150 | 10 | 80 | ns | |
| TRHAV | $\overline{RD}$ Inactive to Next Address Active | TCLCL−45 | | TCLCL−40 | | ns | |
| TCHDTL | Direction Control Active Delay (Note 1) | | 50 | | 50 | ns | |
| TCHDTH | Direction Control Inactive Delay (Note 1) | | 30 | | 30 | ns | |
| TCLGL | GT Active Delay | | 85 | | 50 | ns | |
| TCLGH | GT Inactive Delay | | 85 | | 50 | ns | |
| TRLRH | $\overline{RD}$ Width | 2TCLCL−75 | | 2TCLCL−50 | | ns | |
| TOLOH | Output Rise Time | | 20 | | 20 | ns | From 0.8V to 2.0V |
| TOHOL | Output Fall Time | | 12 | | 12 | ns | From 2.0V to 0.8V |

**NOTES:**
1. Signal at 8284 or 8288 shown for reference only.
2. Setup requirement for asynchronous signal only to guarantee recognition at next CLK.
3. Applies only to T3 and wait states.
4. Applies only to T2 state (8 ns into T3 state).

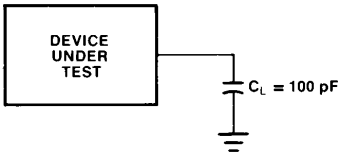
### A.C. TESTING INPUT, OUTPUT WAVEFORM




231456–11

A.C. Testing; Inputs are driven at 2.4V for a logic "1" and 0.45V for a logic "0". Timing measurements are made at 1.5V for both a logic "1" and logic "0".

### A.C. TESTING LOAD CIRCUIT

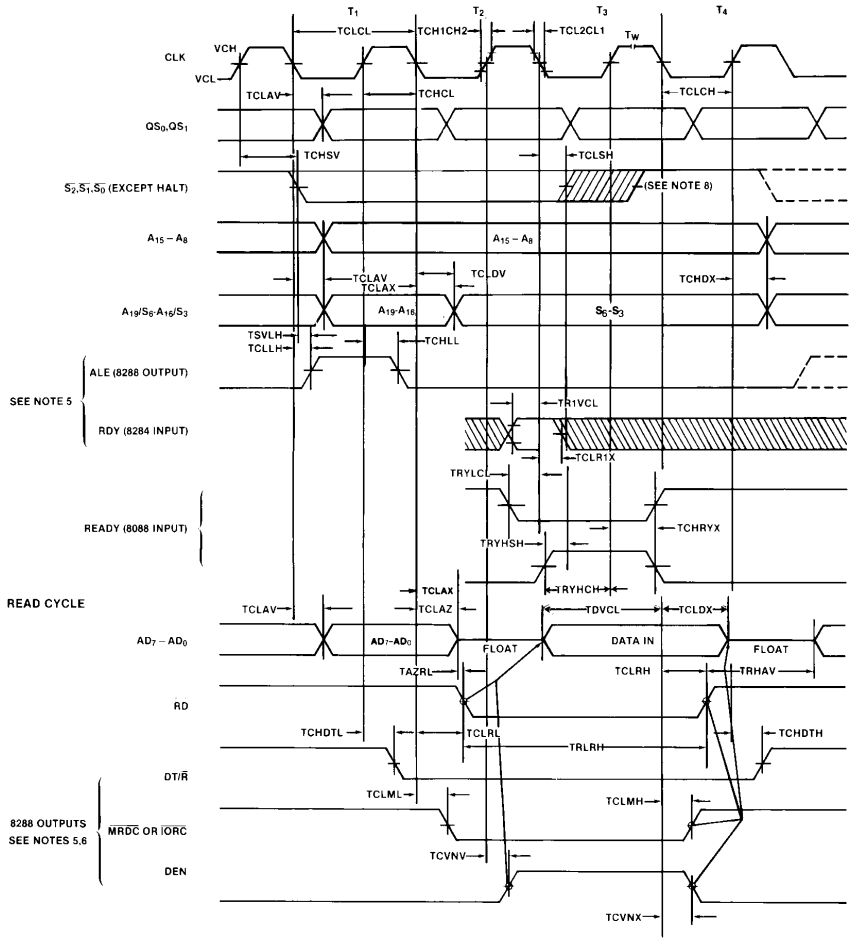



231456–12

$C_L$ Includes Jig Capacitance

## WAVEFORMS (Continued)

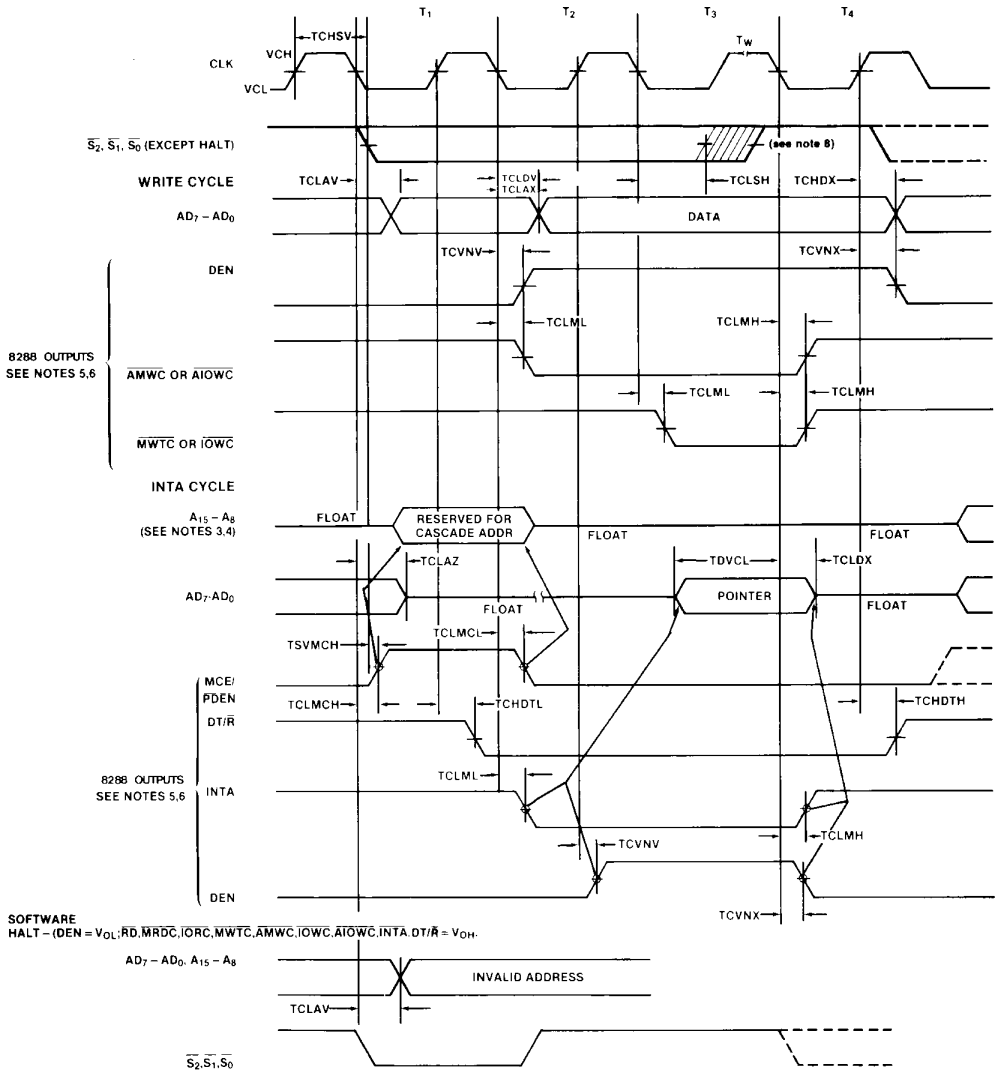
### BUS TIMING—MAXIMUM MODE SYSTEM




231456–15

## WAVEFORMS (Continued)

### BUS TIMING—MAXIMUM MODE SYSTEM (USING 8288)



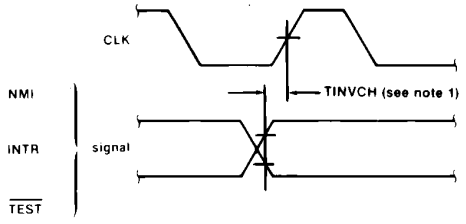

231456–16

**NOTES:**

1. All signals switch between $V_{OH}$ and $V_{OL}$ unless otherwise specified.
2. RDY is sampled near the end of $T_2$, $T_3$, $T_W$ to determine if $T_W$ machines states are to be inserted.
3. Cascade address is valid between first and second INTA cycles.
4. Two INTA cycles run back-to-back. The 8088 local ADDR/DATA bus is floating during both INTA cycles. Control for pointer address is shown for second INTA cycle.
5. Signals at 8284 or 8288 are shown for reference only.
6. The issuance of the 8288 command and control signals ($\overline{MRDC}$, $\overline{MWTC}$, $\overline{AMWC}$, $\overline{IORC}$, $\overline{IOWC}$, $\overline{AIOWC}$, $\overline{INTA}$ and DEN) lags the active high 8288 CEN.
7. All timing measurements are made at 1.5V unless otherwise noted.
8. Status inactive in state just prior to $T_4$.

## WAVEFORMS (Continued)

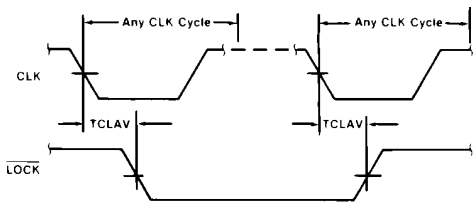
### ASYNCHRONOUS SIGNAL RECOGNITION




231456–17

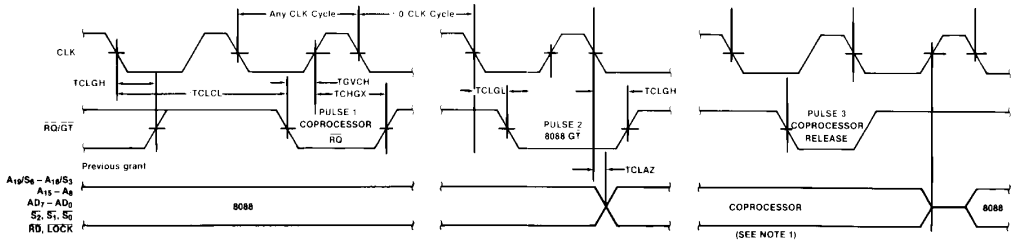
**NOTE:**
1. Setup requirements for asynchronous signals only to guarantee recognition at next CLK.

### BUS LOCK SIGNAL TIMING (MAXIMUM MODE ONLY)




231456–18

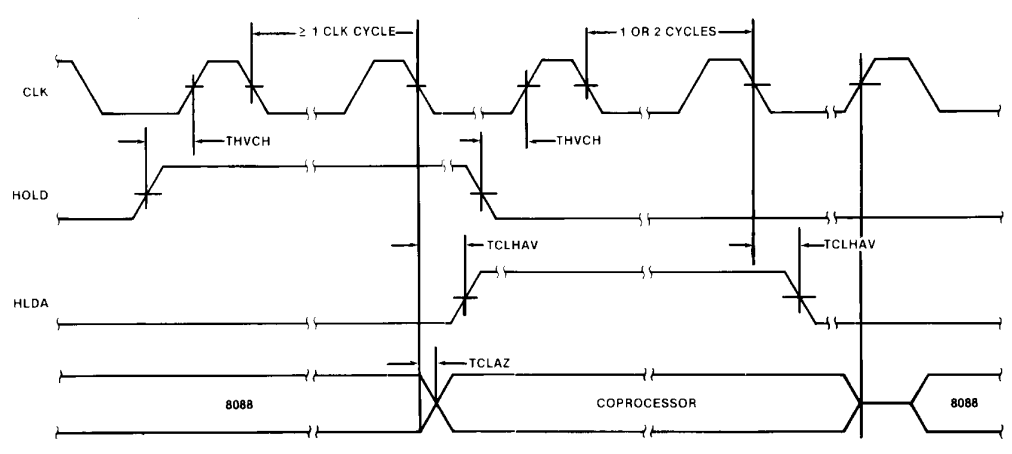
### REQUEST/GRANT SEQUENCE TIMING (MAXIMUM MODE ONLY)




231456–19

**NOTE:**
1. The coprocessor may not drive the busses outside the region shown without risking contention.

### HOLD/HOLD ACKNOWLEDGE TIMING (MINIMUM MODE ONLY)




231456–20

## 8086/8088 Instruction Set Summary

| Mnemonic and Description | Instruction Code | | | |
|---|---|---|---|---|
| **DATA TRANSFER** | | | | |
| **MOV = Move:** | **76543210** | **76543210** | **76543210** | **76543210** |
| Register/Memory to/from Register | 100010dw | mod reg r/m | | |
| Immediate to Register/Memory | 1100011w | mod 000 r/m | data | data if w = 1 |
| Immediate to Register | 1011w reg | data | data if w = 1 | |
| Memory to Accumulator | 1010000w | addr-low | addr-high | |
| Accumulator to Memory | 1010001w | addr-low | addr-high | |
| Register/Memory to Segment Register | 10001110 | mod 0 reg r/m | | |
| Segment Register to Register/Memory | 10001100 | mod 0 reg r/m | | |
| **PUSH = Push:** | | | | |
| Register/Memory | 11111111 | mod 110 r/m | | |
| Register | 01010 reg | | | |
| Segment Register | 000 reg 110 | | | |
| **POP = Pop:** | | | | |
| Register/Memory | 10001111 | mod 000 r/m | | |
| Register | 01011 reg | | | |
| Segment Register | 000 reg 111 | | | |
| **XCHG = Exchange:** | | | | |
| Register/Memory with Register | 1000011w | mod reg r/m | | |
| Register with Accumulator | 10010 reg | | | |
| **IN = Input from:** | | | | |
| Fixed Port | 1110010w | port | | |
| Variable Port | 1110110w | | | |
| **OUT = Output to:** | | | | |
| Fixed Port | 1110011w | port | | |
| Variable Port | 1110111w | | | |
| **XLAT** = Translate Byte to AL | 11010111 | | | |
| **LEA** = Load EA to Register | 10001101 | mod reg r/m | | |
| **LDS** = Load Pointer to DS | 11000101 | mod reg r/m | | |
| **LES** = Load Pointer to ES | 11000100 | mod reg r/m | | |
| **LAHF** = Load AH with Flags | 10011111 | | | |
| **SAHF** = Store AH into Flags | 10011110 | | | |
| **PUSHF** = Push Flags | 10011100 | | | |
| **POPF** = Pop Flags | 10011101 | | | |

## 8086/8088 Instruction Set Summary (Continued)

| Mnemonic and Description | Instruction Code | | | |
|---|---|---|---|---|
| **ARITHMETIC** | 7 6 5 4 3 2 1 0 | 7 6 5 4 3 2 1 0 | 7 6 5 4 3 2 1 0 | 7 6 5 4 3 2 1 0 |
| **ADD = Add:** | | | | |
| Reg./Memory with Register to Either | 0 0 0 0 0 0 d w | mod reg r/m | | |
| Immediate to Register/Memory | 1 0 0 0 0 0 s w | mod 0 0 0 r/m | data | data if s:w = 01 |
| Immediate to Accumulator | 0 0 0 0 0 1 0 w | data | data if w = 1 | |
| **ADC = Add with Carry:** | | | | |
| Reg./Memory with Register to Either | 0 0 0 1 0 0 d w | mod reg r/m | | |
| Immediate to Register/Memory | 1 0 0 0 0 0 s w | mod 0 1 0 r/m | data | data if s:w = 01 |
| Immediate to Accumulator | 0 0 0 1 0 1 0 w | data | data if w = 1 | |
| **INC = Increment:** | | | | |
| Register/Memory | 1 1 1 1 1 1 1 w | mod 0 0 0 r/m | | |
| Register | 0 1 0 0 0 reg | | | |
| **AAA** = ASCII Adjust for Add | 0 0 1 1 0 1 1 1 | | | |
| **BAA** = Decimal Adjust for Add | 0 0 1 0 0 1 1 1 | | | |
| **SUB = Subtract:** | | | | |
| Reg./Memory and Register to Either | 0 0 1 0 1 0 d w | mod reg r/m | | |
| Immediate from Register/Memory | 1 0 0 0 0 0 s w | mod 1 0 1 r/m | data | data if s:w = 01 |
| Immediate from Accumulator | 0 0 1 0 1 1 0 w | data | data if w = 1 | |
| **SSB = Subtract with Borrow** | | | | |
| Reg./Memory and Register to Either | 0 0 0 1 1 0 d w | mod reg r/m | | |
| Immediate from Register/Memory | 1 0 0 0 0 0 s w | mod 0 1 1 r/m | data | data if s:w = 01 |
| Immediate from Accumulator | 0 0 0 1 1 1 w | data | data if w = 1 | |
| **DEC = Decrement:** | | | | |
| Register/memory | 1 1 1 1 1 1 1 w | mod 0 0 1 r/m | | |
| Register | 0 1 0 0 1 reg | | | |
| **NEG** = Change sign | 1 1 1 1 0 1 1 w | mod 0 1 1 r/m | | |
| **CMP = Compare:** | | | | |
| Register/Memory and Register | 0 0 1 1 1 0 d w | mod reg r/m | | |
| Immediate with Register/Memory | 1 0 0 0 0 0 s w | mod 1 1 1 r/m | data | data if s:w = 01 |
| Immediate with Accumulator | 0 0 1 1 1 1 0 w | data | data if w = 1 | |
| **AAS** = ASCII Adjust for Subtract | 0 0 1 1 1 1 1 1 | | | |
| **DAS** = Decimal Adjust for Subtract | 0 0 1 0 1 1 1 1 | | | |
| **MUL** = Multiply (Unsigned) | 1 1 1 1 0 1 1 w | mod 1 0 0 r/m | | |
| **IMUL** = Integer Multiply (Signed) | 1 1 1 1 0 1 1 w | mod 1 0 1 r/m | | |
| **AAM** = ASCII Adjust for Multiply | 1 1 0 1 0 1 0 0 | 0 0 0 0 1 0 1 0 | | |
| **DIV** = Divide (Unsigned) | 1 1 1 1 0 1 1 w | mod 1 1 0 r/m | | |
| **IDIV** = Integer Divide (Signed) | 1 1 1 1 0 1 1 w | mod 1 1 1 r/m | | |
| **AAD** = ASCII Adjust for Divide | 1 1 0 1 0 1 0 1 | 0 0 0 0 1 0 1 0 | | |
| **CBW** = Convert Byte to Word | 1 0 0 1 1 0 0 0 | | | |
| **CWD** = Convert Word to Double Word | 1 0 0 1 1 0 0 1 | | | |

## 8086/8088 Instruction Set Summary (Continued)

| Mnemonic and Description | Instruction Code | | | |
|---|---|---|---|---|
| **LOGIC** | 76543210 | 76543210 | 76543210 | 76543210 |
| **NOT** = Invert | 1 1 1 1 0 1 1 w | mod 0 1 0 r/m | | |
| **SHL/SAL** = Shift Logical/Arithmetic Left | 1 1 0 1 0 0 v w | mod 1 0 0 r/m | | |
| **SHR** = Shift Logical Right | 1 1 0 1 0 0 v w | mod 1 0 1 r/m | | |
| **SAR** = Shift Arithmetic Right | 1 1 0 1 0 0 v w | mod 1 1 1 r/m | | |
| **ROL** = Rotate Left | 1 1 0 1 0 0 v w | mod 0 0 0 r/m | | |
| **ROR** = Rotate Right | 1 1 0 1 0 0 v w | mod 0 0 1 r/m | | |
| **RCL** = Rotate Through Carry Flag Left | 1 1 0 1 0 0 v w | mod 0 1 0 r/m | | |
| **RCR** = Rotate Through Carry Right | 1 1 0 1 0 0 v w | mod 0 1 1 r/m | | |
| **AND** = **And:** | | | | |
| Reg./Memory and Register to Either | 0 0 1 0 0 0 d w | mod reg r/m | | |
| Immediate to Register/Memory | 1 0 0 0 0 0 0 w | mod 1 0 0 r/m | data | data if w = 1 |
| Immediate to Accumulator | 0 0 1 0 0 1 0 w | data | data if w = 1 | |
| **TEST** = **And Function to Flags. No Result:** | | | | |
| Register/Memory and Register | 1 0 0 0 0 1 0 w | mod reg r/m | | |
| Immediate Data and Register/Memory | 1 1 1 1 0 1 1 w | mod 0 0 0 r/m | data | data if w = 1 |
| Immediate Data and Accumulator | 1 0 1 0 1 0 0 w | data | data if w = 1 | |
| **OR** = **Or:** | | | | |
| Reg./Memory and Register to Either | 0 0 0 0 1 0 d w | mod reg r/m | | |
| Immediate to Register/Memory | 1 0 0 0 0 0 0 w | mod 0 0 1 r/m | data | data if w = 1 |
| Immediate to Accumulator | 0 0 0 0 1 1 0 w | data | data if w = 1 | |
| **XOR** = **Exclusive or:** | | | | |
| Reg./Memory and Register to Either | 0 0 1 1 0 0 d w | mod reg r/m | | |
| Immediate to Register/Memory | 1 0 0 0 0 0 0 w | mod 1 1 0 r/m | data | data if w = 1 |
| Immediate to Accumulator | 0 0 1 1 0 1 0 w | data | data if w = 1 | |
| **STRING MANIPULATION** | | | | |
| **REP** = Repeat | 1 1 1 1 0 0 1 z | | | |
| **MOVS** = Move Byte/Word | 1 0 1 0 0 1 0 w | | | |
| **CMPS** = Compare Byte/Word | 1 0 1 0 0 1 1 w | | | |
| **SCAS** = Scan Byte/Word | 1 0 1 0 1 1 1 w | | | |
| **LODS** = Load Byte/Wd to AL/AX | 1 0 1 0 1 1 0 w | | | |
| **STOS** = Stor Byte/Wd from AL/A | 1 0 1 0 1 0 1 w | | | |
| **CONTROL TRANSFER** | | | | |
| **CALL** = **Call:** | | | | |
| Direct Within Segment | 1 1 1 0 1 0 0 0 | disp-low | disp-high | |
| Indirect Within Segment | 1 1 1 1 1 1 1 1 | mod 0 1 0 r/m | | |
| Direct Intersegment | 1 0 0 1 1 0 1 0 | offset-low | offset-high | |
| | | seg-low | seg-high | |
| Indirect Intersegment | 1 1 1 1 1 1 1 1 | mod 0 1 1 r/m | | |

## 8086/8088 Instruction Set Summary (Continued)

| Mnemonic and Description | Instruction Code | | |
|---|---|---|---|
| **JMP = Unconditional Jump:** | 76543210 | 76543210 | 76543210 |
| Direct Within Segment | 11101001 | disp-low | disp-high |
| Direct Within Segment-Short | 11101011 | disp | |
| Indirect Within Segment | 11111111 | mod 100 r/m | |
| Direct Intersegment | 11101010 | offset-low | offset-high |
| | | seg-low | seg-high |
| Indirect Intersegment | 11111111 | mod 101 r/m | |
| **RET = Return from CALL:** | | | |
| Within Segment | 11000011 | | |
| Within Seg Adding Immed to SP | 11000010 | data-low | data-high |
| Intersegment | 11001011 | | |
| Intersegment Adding Immediate to SP | 11001010 | data-low | data-high |
| **JE/JZ** = Jump on Equal/Zero | 01110100 | disp | |
| **JL/JNGE** = Jump on Less/Not Greater or Equal | 01111100 | disp | |
| **JLE/JNG** = Jump on Less or Equal/ Not Greater | 01111110 | disp | |
| **JB/JNAE** = Jump on Below/Not Above or Equal | 01110010 | disp | |
| **JBE/JNA** = Jump on Below or Equal/ Not Above | 01110110 | disp | |
| **JP/JPE** = Jump on Parity/Parity Even | 01111010 | disp | |
| **JO** = Jump on Overflow | 01110000 | disp | |
| **JS** = Jump on Sign | 01111000 | disp | |
| **JNE/JNZ** = Jump on Not Equal/Not Zero | 01110101 | disp | |
| **JNL/JGE** = Jump on Not Less/Greater or Equal | 01111101 | disp | |
| **JNLE/JG** = Jump on Not Less or Equal/ Greater | 01111111 | disp | |
| **JNB/JAE** = Jump on Not Below/Above or Equal | 01110011 | disp | |
| **JNBE/JA** = Jump on Not Below or Equal/Above | 01110111 | disp | |
| **JNP/JPO** = Jump on Not Par/Par Odd | 01111011 | disp | |
| **JNO** = Jump on Not Overflow | 01110001 | disp | |
| **JNS** = Jump on Not Sign | 01111001 | disp | |
| **LOOP** = Loop CX Times | 11100010 | disp | |
| **LOOPZ/LOOPE** = Loop While Zero/Equal | 11100001 | disp | |
| **LOOPNZ/LOOPNE** = Loop While Not Zero/Equal | 11100000 | disp | |
| **JCXZ** = Jump on CX Zero | 11100011 | disp | |
| **INT = Interrupt** | | | |
| Type Specified | 11001101 | type | |
| Type 3 | 11001100 | | |
| **INTO** = Interrupt on Overflow | 11001110 | | |
| **IRET** = Interrupt Return | 11001111 | | |

**8086/8088 Instruction Set Summary** (Continued)

| Mnemonic and Description | Instruction Code | |
|---|---|---|
| | 76543210 | 76543210 |
| **PROCESSOR CONTROL** | | |
| **CLC** = Clear Carry | 11111000 | |
| **CMC** = Complement Carry | 11110101 | |
| **STC** = Set Carry | 11111001 | |
| **CLD** = Clear Direction | 11111100 | |
| **STD** = Set Direction | 11111101 | |
| **CLI** = Clear Interrupt | 11111010 | |
| **STI** = Set Interrupt | 11111011 | |
| **HLT** = Halt | 11110100 | |
| **WAIT** = Wait | 10011011 | |
| **ESC** = Escape (to External Device) | 11011xxx | mod x x x r/m |
| **LOCK** = Bus Lock Prefix | 11110000 | |

**NOTES:**
AL = 8-bit accumulator
AX = 16-bit accumulator
CX = Count register
DS = Data segment
ES = Extra segment
Above/below refers to unsigned value
Greater = more positive:
Less = less positive (more negative) signed values
if d = 1 then "to" reg; if d = 0 then "from" reg
if w = 1 then word instruction; if w = 0 then byte instruction
if mod = 11 then r/m is treated as a REG field
if mod = 00 then DISP = 0*, disp-low and disp-high are absent
if mod = 01 then DISP = disp-low sign-extended to 16 bits, disp-high is absent
if mod = 10 then DISP = disp-high; disp-low
if r/m = 000 then EA = (BX) + (SI) + DISP
if r/m = 001 then EA = (BX) + (DI) + DISP
if r/m = 010 then EA = (BP) + (SI) + DISP
if r/m = 011 then EA = (BP) + (DI) + DISP
if r/m = 100 then EA = (SI) + DISP
if r/m = 101 then EA = (DI) + DISP
if r/m = 110 then EA = (BP) + DISP*
if r/m = 111 then EA = (BX) + DISP
DISP follows 2nd byte of instruction (before data if required)
*except if mod = 00 and r/m = then EA = disp-high: disp-low.
if s:w = 01 then 16 bits of immediate data form the operand
if s:w = 11 then an immediate data byte is sign extended to form the 16-bit operand
if v = 0 then "count" = 1; if v = 1 then "count" in (CL) register
x = don't care
z is used for string primitives for comparison with ZF FLAG
SEGMENT OVERRIDE PREFIX

| 0 0 1 reg 1 1 0 |
|---|

REG is assigned according to the following table:

| 16-Bit (w = 1) | 8-Bit (w = 0) | Segment |
|---|---|---|
| 000 AX | 000 AL | 00 ES |
| 001 CX | 001 CL | 01 CS |
| 010 DX | 010 DL | 10 SS |
| 011 BX | 011 BL | 11 DS |
| 100 SP | 100 AH | |
| 101 BP | 101 CH | |
| 110 SI | 110 DH | |
| 111 DI | 111 BH | |

Instructions which reference the flag register file as a 16-bit object use the symbol FLAGS to represent the file:
FLAGS =
X:X:X:X:(OF):(DF):(IF):(TF):(SF):(ZF):X:(AF):X:(PF):X:(CF)

Mnemonics © Intel, 1978

## DATA SHEET REVISION REVIEW

The following list represents key differences between this and the -005 data sheet. Please review this summary carefully.

1. The Intel 8088 implementation technology (HMOS) has been changed to (HMOS-II).